

法政大学学術機関リポジトリ
HOSEI UNIVERSITY REPOSITORY

粒子群最適化法に基づく動的な最大電力点追従について

著者	明野 龍介
出版者	法政大学大学院理工学研究科
雑誌名	法政大学大学院紀要．理工学・工学研究科編
巻	59
ページ	1-5
発行年	2018-03-31
URL	http://doi.org/10.15002/00021548

粒子群最適化法に基づく動的な最大電力点追従について

PARTICLE SWARM OPTIMIZER BASED DYNAMIC MAXIMUM POWER POINT TRACKING

明野 龍介

Ryusuke AKENO

指導教員 斎藤利通

法政大学大学院理工学研究科電気電子工学専攻修士課程

This paper presents the sensitive particle swarm optimizer (SPSO) for maximum power point tracking in photovoltaic array under partial shading condition. The cost function corresponds to the voltage-versus-power characteristic of the photovoltaic array. Depending on insolation and temperature, the cost function and its MPP vary in a complicated way. In order to track the dynamic MPP, the SPSO includes several strategies: imaginary particle swarm consisting of sampled voltages for real-time operation, a flexible reset method of the past history for adaptation to dynamic environment, and alternation between reference and non-reference to neighbor particles for escape from a trap. Performing numerical experiments for basic artificial problems, the efficiency of the SPSO is confirmed.

Key Words : Particle Swarm Optimizer, Maximum Power Point Tracking

1. はじめに

粒子群最適化法 (Particle Swarm Optimizer: PSO)[1] は、進化計算の一種で鳥や魚などの生物の群行動に基づいた群知能の代表例である。最適値探索法として知られており、多点探索を行うメタヒューリスティックである。PSO は生物を解候補である粒子にモデル化し、各粒子は速度情報、位置情報、探索過程における最良解情報 (P_{best}) を持ち、粒子群の探索過程における最良解情報 (G_{best}) を持つ。粒子はこれらの情報を参照し、解空間を飛び回することで問題に対する近似解を得られる。基本的に簡素なアルゴリズムであり、評価関数の勾配情報 (微分可能) を必要とせず、大域的探索が可能であるという特徴を持つ。このような特徴から様々な問題へ応用されたり [2][3]、本論文では太陽光発電の動的な最大電力点追従 (Dynamic Maximum Power Point Tracking: DMPPT)[4] への応用を考える。この応用では太陽電池の PV 特性を評価関数としてその最大値 (最大電力) を探索していく。太陽電池の PV 特性は、雲や建物の影、温度などといった外部環境に影響を受けやすい。そのため複数の太陽電池を制御する場合、PV 特性は複雑な多峰性関数に変化し複数のピークを有する。また、太陽は時間と共に動くのでリアルタイムで追従を行う場合、評価関数は動的な多峰性関数になる。以上のことから PSO を DMPPT に適用させるにあたり以下のような問題がある。一つは、回路上ではある時刻に動作電圧は一つしかとれないので、複数の動作電圧に対応する複数の粒子の生成は困難であること。二つ目は、最大電力点が現象する場合、各粒子の過去の履歴を代表する情報が悪影響を及ぼすことがある。三つ目は、探索からしばらくすると粒子が収束し探索が停止してしまう事である。これらの問題を解決するために、SPSO[5] を提案する。三つの問題に対応した新しい改良型 PSO アルゴリズムを簡素な問題に適用させ、その性能を検証する。

2. 光電変換システム

太陽電池とは、太陽光エネルギーを電気エネルギーに変換する素子である。光エネルギーを電気エネルギーに変換する効果を光起電力効果という。PSO を適用する評価関数は、太陽電池セルの特性に基づいて構成される。セルの数理モデルとしてセルの PV 特性を数式化する。太陽電池の等価回路を図 1 に示す。

$$i_j = g(v_j, S_j(t)) = I_{ph}(t) - I_{rr} \left(\exp \left(\frac{qv_j}{kAT} \right) - 1 \right) \quad (1)$$

$$I_{ph}(t) = I_{scr} \frac{S_j(t)}{100}, \quad j = 1 \sim N_C$$

ただし、 N_C はセル数であり、 $j \in \{1, \dots, N_C\}$ は複数のセルを区別するための添え字である。また、式中の記号の意味は以下の通りである。

v_j [V]: j 番目のセルの端子電圧。

i_j [A]: j 番目のセルの端子電流。

$S_j(t)$ [mW/cm²]: 日射強度。

$I_{scr} = 2.52$ [A]: 短絡電流。

I_{ph} [A]: 光発生電流。

$I_{rr} = 20$ [μ A]: 逆方向飽和電流。

$q = 1.6 \times 10^{-19}$ [C]: 電気素量。

$k = 1.38$ [J/°K]: ボルツマン定数。

$T = 301$ [°K]: セルの温度。

$A = 1.92$: 理想ダイオード因子。

図 2 のようにセルの一部に影が掛かると、VI 特性は図 2(a) 日射強度 S_j によって変化する。直列結合した結合セルの電圧と各セルの電圧は次式で与えられる。

$$\begin{aligned}
v &= R(i, t) = r_1(i, t) + r_2(i, t) + \cdots + r_{N_C}(i, t) \\
v_1 &= r_1(i_1, t) = g^{-1}(v_1, S_1(t)) \\
v_2 &= r_2(i_2, t) = g^{-1}(v_2, S_2(t)) \\
&\vdots \\
v_{N_C} &= r_{N_C}(i_{N_C}, t) = g^{-1}(v_{N_C}, S_{N_C}(t))
\end{aligned} \tag{2}$$

このときの直列結合したセルの VI 特性を図 2(b) に示す。ただし、 g^{-1} は v_j に関する g の逆関数である。各セルの電流は日射量に依存し、日射量が少ないセルの端子電流が減少する。バイパスダイオードは逆流を防止する役割である。結合セルの PV 特性は次式で与えられる。

$$P = vi = vR^{-1}(v, t) \equiv F(v, t) \tag{3}$$

この PV 特性が評価関数に対応し、電圧が v に対応している。各セルへの日射量が同じ場合には、セルの数がいくつになろうとも単峰性の PV 特性になる。しかし、PV 特性の形状は各セルにあたる日射量に応じて図 2(c) のように多峰性の関数となるとなる。本論文では、 $N_c = 3$ で各セルの部分影が異なり、3 つのピークを持つ場合を考える。日射強度が時不変の場合、評価関数も時不変となり、これを静的評価関数と呼ぶ。また、日射強度が時変だった場合、評価関数も時変となり、これを動的評価関数と呼ぶ。動的評価関数は図 3 のように MPP が時間と共に変化していく。

3. 評価関数

本論文で扱う回路図を図 4 に示す。図のように三つの直列結合された太陽電池セルを二つ並列につなげたものを使用する。各モジュールの端子電圧は以下で表せる。

$$\begin{aligned}
V_1 &= v_{11} + v_{12} + v_{13} \\
v_2 &= v_{21} + v_{22} + v_{23} \\
v_{11} &= r_{11}(i_{11}, t) = g^{-1}(v_{11}, S_{11}(t)) \\
&\vdots \\
v_{mN_c} &= r_{mN_c}(i_{mN_c}, t) = g^{-1}(v_{mN_c}, S_{mN_c}(t))
\end{aligned} \tag{4}$$

ここで、 m はモジュールの数を表す。

結合モジュールの PV 特性は以下の式で表される。

$$P = (V_1, V_2) = V_1 I_1 + V_2 I_2 = F(V, t) \tag{5}$$

この PV 特性が評価関数に対応し、電圧が V に対応している。また、評価関数である PV 特性と解空間との関係性を図 5 に示す。

4. 仮想粒子

ここでは仮想粒子を定義する。問題点のひとつである、回路上ではある時刻では動作電圧は一つの値しかとれないという問題を解決するために適当な時間間隔によるサンプル値を仮想粒子として用いる手法を提案する。PSO は複数の粒子の相互作用によって最適値（解）を探索する。これを近似的に実現するため、過去 M 個の x のサンプル値を M 個の仮想粒子位置とし、これによって仮想粒子群を構成する。粒子位置の番号とサンプル時刻を対応づけるため、 M 個のサンプルごとに番号をつける。 $M = 10$ の場合は以下ようになる。

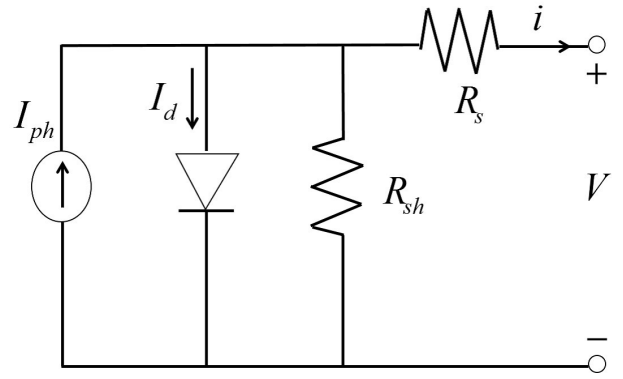


図 1 太陽電池等価回路

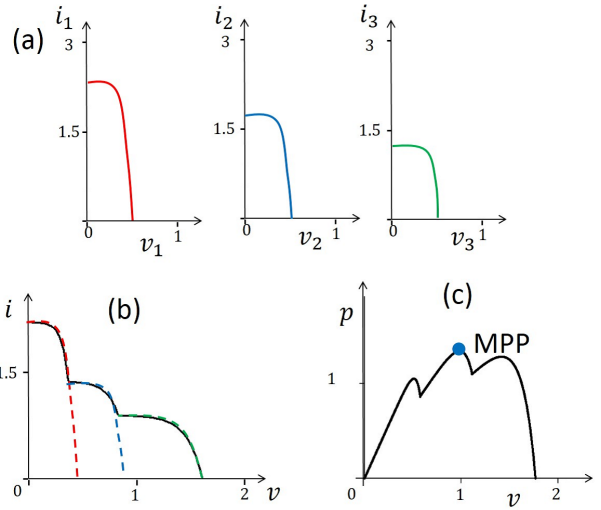
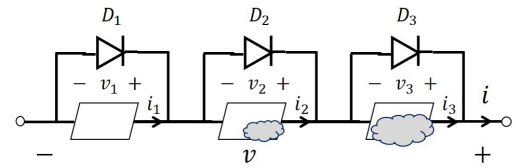


図 2 直列接続したセル。(a) 各セルの電流電圧特性、(b) 電流電圧特性、(c) 電力電圧特性

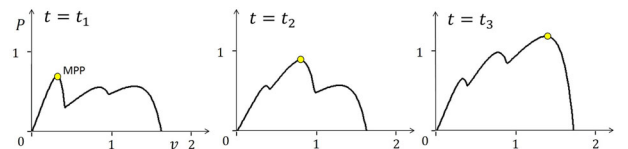


図 3 動的な PV 特性

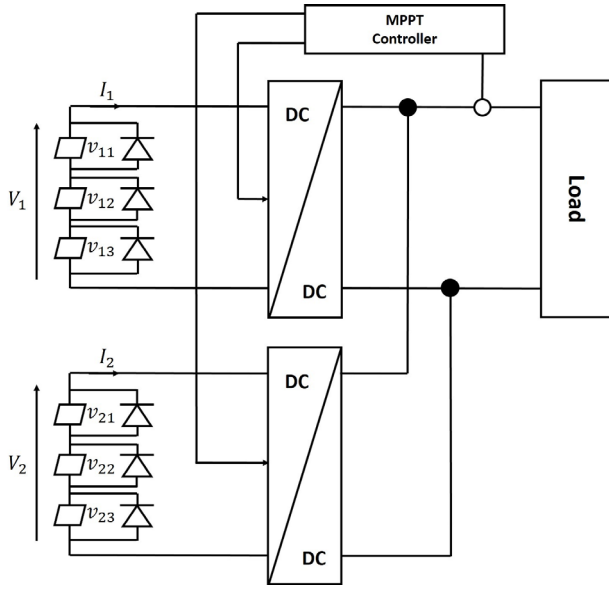


図4 2次元のMPPT制御回路

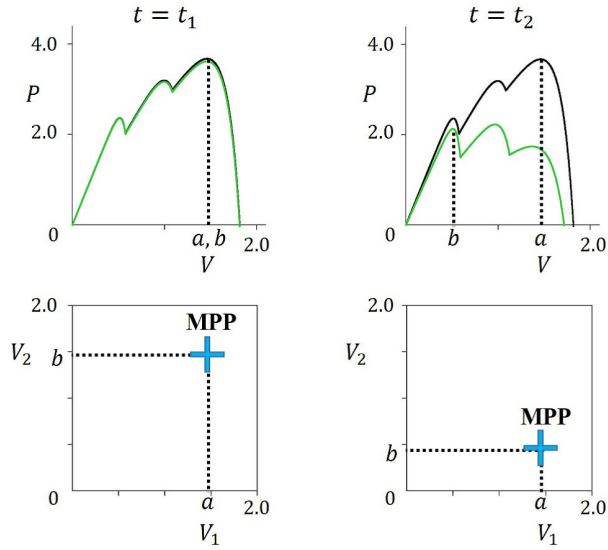


図5 評価関数と解空間の関係性

$$\begin{array}{lll}
 1 \leq n \leq 10 & 11 \leq n \leq 1 & \cdots & N_a - 9 \leq n \leq N_a \\
 X_1 = x(1) & X_1 = x(6) & \cdots & X_1 = x(N_a - 9) \\
 X_2 = x(2) & X_2 = x(7) & \cdots & X_2 = x(N_a - 8) \\
 X_3 = x(3) & X_3 = x(8) & \cdots & X_3 = x(N_a - 7) \\
 \vdots & \vdots & \cdots & \vdots \\
 X_{10} = x(10) & X_{10} = x(20) & \cdots & X_{10} = x(N_a)
 \end{array}$$

ただし、 N_a は M の倍数とする。 X_i を i 番目の仮想粒子位置とよぶ。以下、簡単のため、 $M = 10$ として説明する。10個の X_i と仮想粒子速度 Y_i で仮想粒子群を構成する。

$$P_i = (X_i, Y_i), i = 1 \sim 10 \quad (6)$$

以下では、仮想粒子位置、仮想粒子速度、仮想粒子群を各々粒子位置、粒子速度、粒子群とよぶ。粒子の結合法には様々なものがあるが、本論文ではリング結合を用いる。 P_i の近傍は $\{P_{i-1}, P_i, P_{i+1}\}$ となる。ただし、 $i = 1 \sim 10$, $P_{-1} \equiv P_{10}$, $P_{11} \equiv P_1$ である。ここで、パーソナルベスト (Pbest)、ローカルベスト (Lbest)、グローバルベスト (Gbest) を定義する。

i 番目の粒子で、現在までに最も評価の高い値を与えた位置を Pbest 位置とよび、 X_{P_i} であらわす。Pbest 位置における評価値が Pbest であり、 $Pb_i = f(X_{P_i}, n)$ であらわす。ただし、時変評価関数の場合、 n は、その Pbest を与えた時刻とする。

i 番目の粒子の近傍の Pbest 位置で最も評価の高い値を与える位置を Lbest 位置とよび、 X_{L_i} であらわす。Lbest 位置における評価値が Lbest であり、 Lb_i であらわす。

全粒子中最も評価の高い粒子位置を Gbest 位置とよび、 X_G であらわす。その値が Gbest であり、 Gb であらわす。

ここで、Lbest は近傍内の粒子と比較して決定され、近傍は粒子の結合形態に依存することに注意する。リング結合では、粒子 $P_i = (X_i, Y_i)$ の近傍は $\{P_{i-1}, P_i, P_{i+1}\}$ である。これは、合成セルの端子電圧 v のサンプルでは、 $x(n) = X_i$ の近傍は $(x(n-1), x(n), x(n-9))$ と対応する。すなわち、時刻 n のサンプル $x(n)$ と、1 時刻まえのサンプル $x(n-1)$ 、9 時刻前のサンプル $x(n-9)$ の評価に基づいて Lbest が決まる。サンプル間隔に関する検討は、評価関数の形に依存するため特定のサンプル間隔を断定することは難しい。しかし、すべてのサンプル値を参照するのは大域探索が出来ず局所解に陥る可能性が高くなるので好ましくない。本論文では、試行錯誤によって決定した。

5. SPSO アルゴリズム

SPSO (Sensitive Particle Swarm Optimizer) は粒子の過去の最良値である Pbest を保持せず、粒子の更新の際に使用しないアルゴリズムである。通常の PSO (Local PSO: LPSO) では Pbest を常に保持し続けるため、多峰性関数のように複数のピークがあり、さらに最良値が時間と共に変化する動的評価関数に適用すると局所解に陥る可能性が高くなる。また、PSO の利点の一つである粒子の収束の速さが動的評価関数では悪手なる場合が存在する。これは、粒子が収束することで探索が停止してしまう可能性があるということである。よって、これらの問題を解決をする為に Pbest を保持せず、更新式にも Pbest を参照しない SPSO を提案する。

以下に SPSO のアルゴリズムを示す。

Step 1 (初期化):

粒子を $x(1)$ から $x(10)$ で初期化する。同時に Lbest も初期化する。

Step 2 (Lbest の更新):

$$\begin{aligned}
 Lb_i &\leftarrow X_j(n) & \text{if } X_j(n) > Lb_i, \quad j \in \{i-1, i, i+1\} \\
 Lb_i &\leftarrow Lb_i & \text{otherwise}
 \end{aligned} \quad (7)$$

Step 3 (位置と速度の更新):

$$\begin{aligned}
 Y(n) &\leftarrow WY(n) + C(Lb_i - X(n)) \\
 X(n) &\leftarrow X(n) + Y(n)
 \end{aligned} \quad (8)$$

ここで W と C は定数パラメータである。以下にパラメータの値を示す。

$$C = 1.4, W = 0.7$$

このパラメータ値は試行錯誤の上、決定した。

Step 4 (Lbest のリセット):

$$\begin{aligned} Lb_i &\leftarrow 0 \\ F_{Lbi} &\leftarrow 0 \end{aligned} \quad (9)$$

Step 5: Let $n \leftarrow n + 1$, let $i \leftarrow n \bmod 10$, まで更新し, $n_{max} = 500$ まで Step 2 から繰り返す.

6. 数値実験

今回の実験に用いたパラメータは以下のように設定した. 粒子数 $M = 10$, 最大探索ステップ数 $n_{max} = 500$, SPSO の探索過程をスナップショットしたものを図 6 に, SPSO の探索過程を図 7 に示す. また, 比較のために LPSO を適用した場合の結果を, スナップショットを図 9, 探索過程は図 10 に示す.

LPSO と SPSO を比較すると, LPSO は最初の MPP には追従できているが, P_{best} に影響を受けその後の MPP の変化に対応できず追従出来ていない. また, 粒子も収束してしまい探索が停止している. しかし, SPSO は粒子がばらけることなく常に MPP 周辺を探索しており, MPP に対して追従していること確認できる.

これは, P_{best} を保持せず参照しないことで粒子が過去の情報に影響を受けることなく, 常に現在の情報のみを参照して更新することによるものが有効であることを示している. また, P_{best} を参照しないことで様々な情報を参照することになった. これによって粒子の収束の速度が制限され, 粒子が集まりにくく, かつ, 散らばりすぎない探索を行っているのて局所解からの脱出に対しても有効であることが確認できた.

ここで, この二つの実験結果を評価するために, 瞬時追従効率 (instantaneous tracking efficiency: ITE), 平均追従効率 (average tracking efficiency: ATE) を定義する.

$$\text{ITE}(n) = \frac{f(X(n), n)}{\text{MPP}} \times 100 \quad [\%] \quad (10)$$

$$\text{ATE} = \frac{1}{n_{max}} \sum_{n=0}^{n_{max}} \text{ITE}(n) \quad [\%] \quad (11)$$

以上の定義式から得られた ATE は, LPSO: 86.9, SPSO = 91.4 となった. また, SPSO の ITE を図 8, LPSO の ITE を図 11 に示す. ATE の比較から LPSO よりも SPSO の方が追従効率が良いことが確認できた.

7. まとめ

P_{best} の情報を保持せず更新式で参照しない SPSO を提案した. 提案した手法を動的多峰性評価関数を用いた DMPPT に適用し, その性能を比較, 検証した. 提案手法により問題となっていた過去の情報が悪影響を及ぼす, 粒子が収束し探索が停止する問題に対して二次元の動的多峰性関数への有効性を示した. また, DMPPT に関しても提案手法によって追従効率の増加が確認できた.

今後の課題は, 探索過程の詳細な解析, 回路実装への検討等が挙げられる.

参考文献

- 1) J. Kennedy and R. Eberhart, Particle Swarm Optimization, Proc of IEEE/ICNN, pp.1942-1948, 1995
- 2) K. Maruyama and T. Saito, Collision Particle Swarm Optimizers and Exploring Periodic Points, NOLTA, IEICE, 5, 4, pp. 423-534, 2014
- 3) S.-T. Hsieh, T.-Y. Sun, C.-L. Lin, and C.-C. Liu, Effective learning rate adjustment of blind source separation based on an improved particle swarm optimizer, IEEE Trans. Evol. Comput., 12, 2. pp. 242-251, 2008.
- 4) R. Akeno, K. Tanakajima, and T. Saito, The Imaginary Particle Swarm Optimizer for Dynamic Maximum Power Point Tracking, Proc. IEEE/SMC, pp. 2978-2983, 2016.
- 5) R. Akeno and T. Saito, Evolutionary Computation based Dynamic Maximum Power Point Tracking, Proc. of NOLTA, pp. 494-497, 2017.

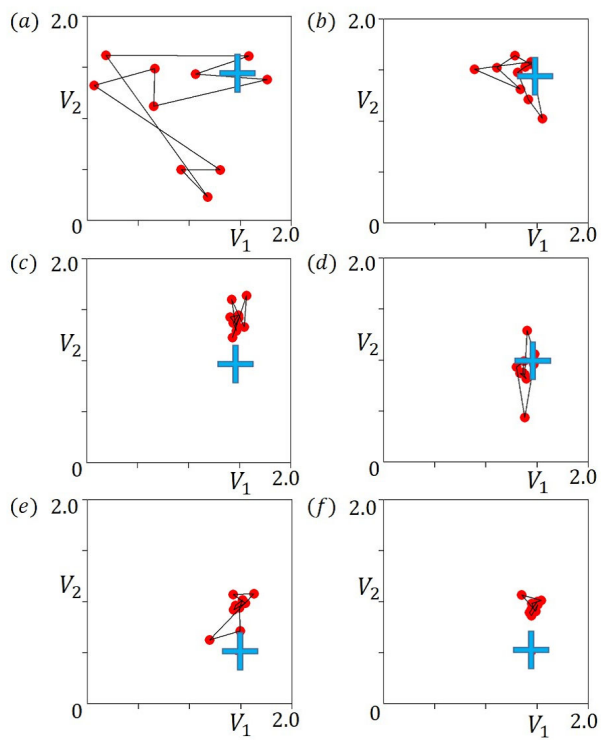


図 6 SPSO のスナップショット

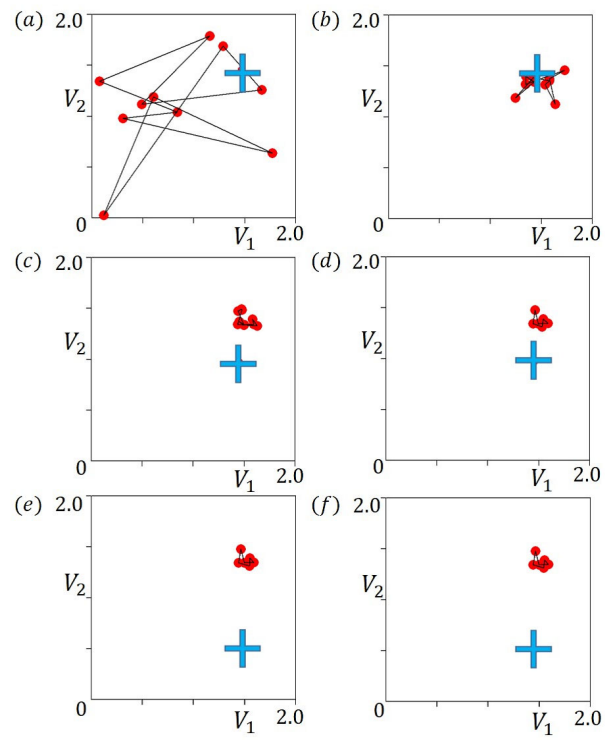


図 9 LPSO のスナップショット

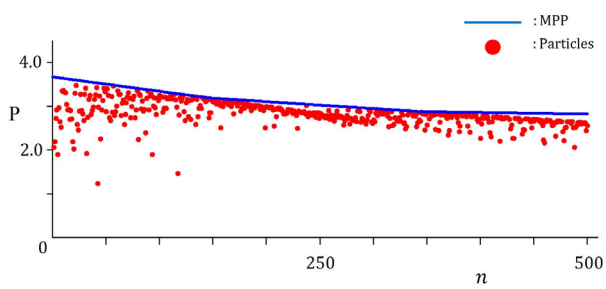


図 7 SPSO の探索過程

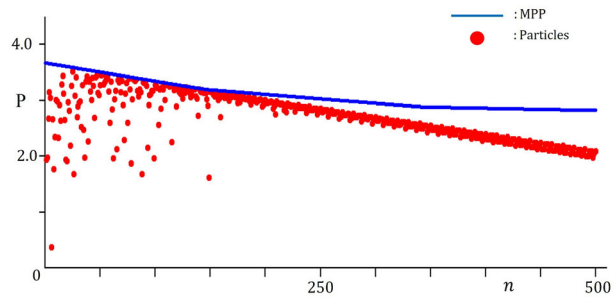


図 10 LPSO の探索過程

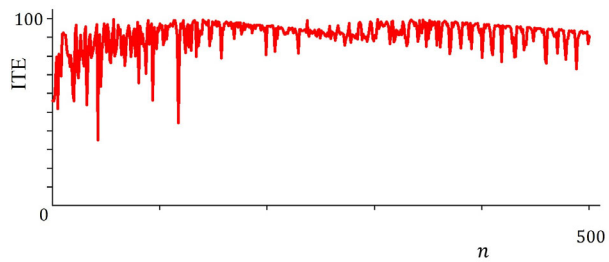


図 8 SPSO の ITE

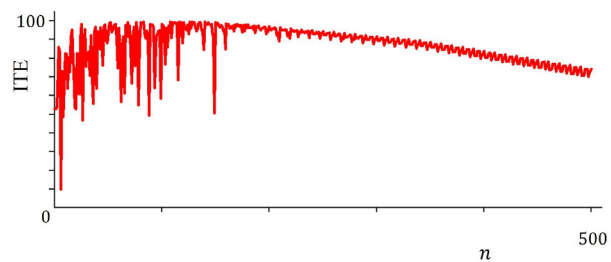


図 11 LPSO の ITE